Model Training Basics

In this section, we will get into more detail on the steps involved in training a model, and then we'll get some hands-on practice with the process in the upcoming lab.

Remember that our ultimate goal is to produce a model we can use to make predictions. Put another way, we want to be able to give the model a set of input features, X, and have it predict the value of some output feature, y.

Parameters and Hyperparameters

When we train a model, a large part of the process involves learning the values of the *parameters* of the model. For example, earlier we looked at the general form for linear regression:

$$y = B_0 + B_1 * x_1 + B_2 * x_2 + B_3 * x_3 ... + B_n * x_n$$

The coefficients in this equation, $B_0 \dots B_n$, determine the intercept and slope of the regression line. When training a linear regression model, we use the training data to figure out what the value of these *parameters* should be. Thus, we can say that *a major goal of model training is to learn the values of the model parameters.*

In contrast, some model parameters are *not* learned from the data. These are called **hyperparameters** and their values are set before training. Here are some examples of hyperparameters:

- The number of layers in a deep neural network
- The number of clusters (such as in a k-means clustering algorithm)
- The learning rate of the model

We must choose some values for these hyperparameters, but we do not necessarily know what the best values will be prior to training. Because of this, a common approach is to take a best guess, train the model, and then tune adjust or *tune* the hyperparameters based on the model's performance.

Splitting the Data

As mentioned in the video, we typically want to split our data into three parts:

- **Training data**
- **Validation data**
- **Test data**

We use the **training data** to learn the values for the *parameters*. Then, we check the model's performance on the **validation data** and *tune* the hyperparameters until the model performs well with the validation data. For instance, perhaps we need to have more or fewer layers in our neural network. We can adjust this hyperparameter and then test the model on the validation data once again to see if its performance has improved.

Finally, once we believe we have our finished model (with both parameters and hyperparameters optimized), we will want to do a final check of its performance—and we need to do this on some fresh **test data** that we did not use during the training process.

QUIZ QUESTION

Let's review the terms we just introduced. See if you can match each one with the correct description.

Submit to check your answer choices!

DESCRIPTION	TERM
Data used to tune the values of the *hyperparameters*.	Validation data
Variables whose value is not learned during training, but rather set as a "best guess" and then tuned.	Hyperparameters
Data used to learn the values of the *parameters*.	Training data
Variables whose value is learned during training.	Parameters
Data used to check the performance of the final, fully trained model.	Test Data

SUBMIT

NEXT